WHAT IS CLAIMED IS:

1. A method for providing printer recognition and management of a print job entity, comprising:

establishing a repository of attributes and status information associated with each print job that passes through a printer system; and

providing an interface to a plurality of components to allow access to the attributes and status information in the repository by the plurality of components.

2. The method of claim 1 wherein the interface comprises at least one of a Web Page channel, a multiplexer to manage the routing of jobs to the print engine and a spooler, a job control function interface, a pipeline interface, an operations panel interface and a pull print interface.

3. The method of claim 1 further comprising providing by the interface an ability for components to process a job according to requirements of the component and reporting job attributes and processing status of the job for common access by other components.

4. The method of claim 1 further comprising providing by the interface access to maintained job variable to the components.

5. The method of claim 1 further comprising providing by the interface to a component access to common variables, the components presenting job attributes or status to the interface.

6. The method of claim 5 wherein the attributes are presented according to requirements dictated by the interface

7. The method of claim 1 wherein the interface provides the ability for components to create job entries, obtain and set job attributes, manipulate the state and status of jobs in the system, and obtain job ordering information pertinent to the calling component.

8. The method of claim 1 wherein the repository provides a global view of jobs within the printer, the global view includes an actively printing job, jobs in the process of being spooled, jobs on the spool queue, and jobs on the pull print queue.

9. The method of claim 1 wherein the interface accommodates either implementation of port connection managers and pass job information from a port connection manager to the repository.

10. The method of claim 1 wherein the interface cancels jobs.

11. The method of claim 10 wherein a cancelled job comprises a current job.

12. The method of claim 10 wherein a cancelled job comprises a job having a selected attribute.

13. The method of claim 1 further comprising providing logical views to obtain a next job to be processed by a component and to obtain a list of all jobs in the order that they are processed.

14. The method of claim 1 further comprises establishing a job monitor for obtaining a Job ID, performing a query for attributes of a job, updating job attributes, canceling jobs, providing logical views of a job, handling printer events, getting attributes of the printer and setting printer attributes.

15. The method of claim 14 wherein the attributes are updated through the job monitor.

16. The method of claim 14 wherein the job monitor provides the ability for any component to set job attributes.

17. The method of claim 14 wherein the job monitor uses job states to control the flow of jobs.

18. The method of claim 14 further comprising responding by the job monitor to a component call, wherein the job monitor determines a next job to process and wherein the component determines valid states for a call.

19. The method of claim 18 further comprising maintaining a valid state for a multiplexer.

20. The method of claim 19 wherein the maintaining a valid state for a multiplexer further comprises:

placing an incoming job into an unknown state when a job identification is requested;

placing the incoming job in the Pull Print queue when the job is stop-flowed at a port connection manager waiting for access to the printer because a print engine is processing another job; and

selecting the incoming job and processing the job according to whether the job must be spooled, may spool or must print.

21. The method of claim 20 wherein the incoming job is routed to the print engine or the spooler according to which comes first when the job is a job that may spool.

22. The method of claim 20 wherein the incoming job is placed in a pending spooler when the job is a job that must be spooled.

23. The method of claim 20 further comprising indicating a done state for the multiplexer when the job has been printed.

24. The method of claim 18 further comprising maintaining a valid state for a spooler.

25. The method of claim 24 wherein the maintaining a valid state for a spooler further comprises:

receiving a job identification request;

entering a not spooled state when the spooler has not yet processed the job;

entering a spooling, can despool state when the job is being written to the spool device thereby allowing the job to be selected for despooling at any time;

entering a spooling, despooling state when the job is being written to the spool device and is also being read from the spool device;

entering a waiting to despool state when the end of the job has been received;

entering a despooling state when the job is being read from the spool device and written to the multiplexer; and

entering the done state when the job is finished being processed by the spooler.

26. The method of claim 25 wherein a job that is printed directly and not processed by the spooler remains in the not spooled state.

27. The method of claim 18 further comprising maintaining a valid state for an interpreter.

28. The method of claim 27 wherein the maintaining a valid state for a interpreter further comprises:

entering a waiting for data stated when job processing by the interpreter has started;

entering an interpreting state when the job is being processed by the interpreter; and

entering a done state when the job is finished being processed by the interpreter.

29. The method of claim 18 further comprising maintaining a valid state for a print engine.

30. The method of claim 29 wherein the maintaining a valid state for a print engine further comprises:

entering a waiting for pages state when job processing by an interpreter has not yet started;

entering a waiting for pages state when the job has started;

entering the pages queued state when one or more pages for the job have been created by the interpreter and written to the page buffer;

entering the pages printing state when one or more pages for the job have been delivered to the output tray; and

entering the done state when the last page for the job has been delivered to the output tray.

31. The method of claim 1 further comprising handling incoming jobs with a port connection manager, wherein the port connection manager calls to a multiplexer to process the job.

32. The method of claim 1 further comprising deciding whether to assign a job to the printer, whether to assign a job to a spooler, whether the job must wait for available resources or whether the job cannot be processed.

33. The method of claim 1 further comprising requesting from a job monitor a job identification prior to processing the job by a multiplexer.

34. The method of claim 33 further comprising storing the job identification in a job table and clearing the job identification from the table when an end of job is called by a port connection manager.

35. The method of claim 1 further comprising providing a job monitor to fetch jobs in an order that is dependent upon the calling component.

36. The method of claim 35 further comprising examining by the job monitor process job states and variables to determine the correct response and to return an appropriate job identification for a job.

37. The method of claim 1 further comprising providing an event registration to provide a methodology for a controller to indicate events to a job monitor, wherein the Job Monitor serves as the system focal point for tracking job related events as they occur during the course of an entire print process.

38. The method of claim 37 further comprising defining events for the job monitor.

39. The method of claim 1 further comprising providing a job monitor for addressing job processing complexity by viewing a job on a higher conceptual plane rather than managing a collection of attributes and status variables that is unique for each data channel.

40. The method of claim 1 further comprising providing a job monitor for providing a common method of accessing the variables associated with a job for the components.

41. An apparatus for providing printer recognition and management of a print job entity, comprising:

a repository of attributes and status information associated with each print job that passes through a printer system; and

an interface to a plurality of components, the interface providing access to the attributes and status information in the repository by the plurality of components.

42. The apparatus of claim 41 wherein the interface comprises at least one of a Web Page channel, a multiplexer to manage the routing of jobs to the print engine and a spooler, a job control function interface, a pipeline interface, an operations panel interface and a pull print interface.

43. The apparatus of claim 41 wherein the interface provides an ability for components to process a job according to requirements of the component and reports job attributes and processing status of the job for common access by other components.

44. The apparatus of claim 41 wherein the interface provides access to maintained job variable to the components.

45. The apparatus of claim 41 wherein the interface provides a component access to common variables, the components presenting job attributes or status to the interface.

46. The apparatus of claim 45 wherein the attributes are presented according to requirements dictated by the interface

47. The apparatus of claim 41 wherein the interface provides the ability for components to create job entries, obtain and set job attributes, manipulate the state and status of jobs in the system, and obtain job ordering information pertinent to the calling component.

48. The apparatus of claim 41 wherein the repository provides a global view of jobs within the printer, the global view includes an actively printing job, jobs in the process of being spooled, jobs on the spool queue, and jobs on the pull print queue.

49. The apparatus of claim 41 wherein the interface accommodates either implementation of port connection managers and pass job information from a port connection manager to the repository.

50. The apparatus of claim 41 wherein the interface cancels jobs.

51. The apparatus of claim 50 wherein a cancelled job comprises a current job.

52. The apparatus of claim 50 wherein a cancelled job comprises a job having a selected attribute.

53. The apparatus of claim 41 wherein the a repository and interface are provided by a job monitor, the job monitor further providing logical views to obtain a next job to be processed by a component and to obtain a list of all jobs in the order that they are processed.

54. The apparatus of claim 41 wherein the job monitor obtains a Job identification, performs a query for attributes of a job, updates job attributes, cancels jobs, provides logical views of a job, handles printer events, gets attributes of the printer and sets printer attributes.

55. The apparatus of claim 54 wherein the attributes are updated through the job monitor.

56. The apparatus of claim 54 wherein the job monitor provides the ability for any component to set job attributes.

57. The apparatus of claim 54 wherein the job monitor uses job states to control the flow of jobs.

58. The apparatus of claim 54 wherein the job monitor responds to a component call, determines a next job to process, the component determining valid states for a call.

59. The apparatus of claim 58 further comprising a multiplexer.

60. The apparatus of claim 59 wherein the valid states for a multiplexer further comprise:

an unknown stated for when a job identification is requested; and

a pull print queue state for the job when the job is stop-flowed at a port connection manager waiting for access to the printer because a print engine is processing another job;

wherein the multiplexer receives the job and selects to place the job in a job must be spooled state, a may spool state or must print state.

61. The apparatus of claim 60 wherein the multiplexer routes the incoming job to the print engine or the spooler according to which becomes available first when the job is a job that may spool.

62. The apparatus of claim 60 wherein the multiplexer places an incoming job in a pending spooler when the job is a job that must be spooled.

63. The apparatus of claim 60 wherein the multiplexer enters a done state for the multiplexer when the job has been printed.

64. The apparatus of claim 58 further comprising a spooler.

65. The apparatus of claim 64 wherein the spooler receiving a job identification request, enters a not spooled state when the spooler has not yet processed the job, enters a spooling, can despool state when the job is being written to the spool device thereby allowing the job to be selected for despooling at any time, enters a spooling, despooling state when the job is being written to the spool device and is also being read from the spool device, enters a waiting to despool state when the end of the job has been received, enters a despooling state when the job is being read from the spool device and written to the multiplexer and enters the done state when the job is finished being processed by the spooler.

66. The apparatus of claim 65 wherein a job that is printed directly and not processed by the spooler remains in the not spooled state.

67. The apparatus of claim 58 further comprising an interpreter.

68. The apparatus of claim 67 wherein the interpreter enters a waiting for data stated when job processing by the interpreter has started, enters an interpreting state when the job is being processed by the interpreter and enters a done state when the job is finished being processed by the interpreter.

69. The apparatus of claim 58 further comprising a print engine.

70. The apparatus of claim 69 wherein the print engine enters a waiting for pages state when job processing by an interpreter has not yet started, enters a waiting for pages state when the job has started, enters the pages queued state when one or more pages for the job have been created by the interpreter and written to the page buffer, enters the pages printing state when one or more pages for the job have been delivered to the output tray and enters the done state when the last page for the job has been delivered to the output tray.

71. The apparatus of claim 41 wherein the a repository and interface are provided by a job monitor, the job monitor further handling incoming jobs with a port connection manager, wherein the port connection manager calls to a multiplexer to process the job.

72. The apparatus of claim 41 wherein the a repository and interface are provided by a job monitor, the job monitor further deciding whether to assign a job to the printer, whether to assign a job to a spooler, whether the job must wait for available resources or whether the job cannot be processed.

73. The apparatus of claim 41 wherein the a repository and interface are provided by a job monitor, the job monitor receiving a request for a job identification prior to processing the job by a multiplexer.

74. The apparatus of claim 73 wherein the job identification is stored in a job table, the job monitor clearing the job identification from the table when an end of job is called by a port connection manager.

75. The apparatus of claim 41 further comprising a job monitor to fetch jobs in an order that is dependent upon the calling component.

76. The apparatus of claim 75 further comprising a job monitor for examining process job states and variables to determine the correct response and to return an appropriate job identification for a job.

77. The apparatus of claim 41 further comprising a job monitor for serving as a focal point for tracking job related events as they occur during the course of an entire print process.

78. The apparatus of claim 77 further comprising events definitions for the job monitor.

79. The apparatus of claim 41 further comprising a job monitor for addressing job processing complexity by viewing a job on a higher conceptual plane rather than managing a collection of attributes and status variables that is unique for each data channel.

80. The apparatus of claim 41 further comprising a job monitor for providing a common method of accessing the variables associated with a job for the components.

81. An article of manufacture comprising a program storage medium readable by a computer, the medium tangibly embodying one or more programs of instructions executable by the computer to perform a method for providing printer recognition and management of a print job entity, the method comprising:

establishing a repository of attributes and status information associated with each print job that passes through a printer system; and

providing an interface to a plurality of components to allow access to the attributes and status information in the repository by the plurality of components.